Listing of Claims:

1~22 (Canceled).

23. (Previously Presented) A conversational browser, comprising:

a command and control interface for interpreting a user command and for generating a request to access one of a CML (conversational markup language) file and a CML application, wherein CML comprises meta-information implementing a conversational dialog to enable interaction with the user in a plurality of user interface modalities; and

a CML processor for parsing and interpreting one of a CML file and CML application to to render the conversational dialog in one or more of the plurality of user interface modalities.

24. (Original) The conversational browser of claim 23, wherein the conversational browser executes on top of a platform that processes input/output (I/O) events under the control of the conversational browser to provide a conversational user interface in accordance with the rendered conversational dialog.

25. (Original) The conversational browser of claim 24, wherein the platform is a conversational virtual machine.

26. (Original) The conversational browser of claim 23, further comprising a dialog manager for managing and controlling the conversational dialog.

27. (Original) The conversational browser of claim 26, wherein the dialog manager allocates conversational engines and conversational data files for rendering the conversational dialog comprising one of the CML file and the CML application and for processing input user commands.

2

28. (Original) The conversational browser of claim 27, wherein the dialog manager is one of controlled and programmed by meta-formation comprising one of a CML file and CML application.

29. (Original) The conversational browser of claim 27, wherein the conversational engines comprise a speech recognition engine, a speaker recognition engine, a text-to-speech engine, a natural language understanding engine, a natural language generation engine, a dual tone multifrequency engine, a speech compression engine, and a combination thereof.

30. (Previously Presented) The conversational browser of claim 23, where the user command can be input in the one or more user interface modalities.

31. (Original) The conversational browser of claim 23, wherein CML is implemented using a declarative format.

32. (Canceled)

33. (Original) The conversational browser of claim 31, wherein the declarative format comprises an extensible markup language (XML) format.

34. (Original) The conversational browser of claim 23, wherein CML is implemented using a combination of declarative and imperative components

35. (Original) The conversational browser of claim 23, wherein the conversational browser implements a "what you hear is what you can say" dialog format.

36. (Original) The conversational browser of claim 23, wherein the conversational browser implements a "say what you heard" dialog format.

37. (Original) The conversational browser of claim 23, wherein the conversational browser implements a "say what you will hear" dialog format.

38. (Original) The conversational browser of claim 23, wherein the conversational browser implements a mixed initiative dialog format.

39. (Previously Presented) A system for accessing information, comprising:

a content server comprising one of content pages, applications, and a combination thereof, wherein the content pages and applications are implemented using a conversational markup language (CML) to describe a conversational dialog for interaction with a user in a plurality of user interface modalities;

a conversational browser for processing one of a CML page and CML application received from the content server to render its conversational dialog in one or more of the plurality of user interface modalities; and

a platform for processing input/output (I/O) events under the control of the conversational browser to provide a conversational user interface in accordance with the rendered conversational dialog.

40. (Previously Presented) The system of claim 39, wherein the system comprises a IVR (interactive voice response) system, wherein a DTMF (dual tone multifrequency) application associated with the IVR system is implemented in CML.

41. (Original) The system of claim 40, wherein the content server comprises the IVR system, and wherein the IVR system is accessible by the conversational browser over a packet-switched network using a standard network protocol.

42. (Original) The system of claim 40, wherein the conversational browser and content server comprise the IVR system, and wherein the IVR system is accessible over a packet switched network using standard network protocol.

43. (Original) The system of claim 40, wherein the IVR system comprises a gateway for capturing and compressing speech, and shipping the compressed speech to the content server comprising the IVR system for processing.

44. (Original) The system of claim 40, wherein the input modality of the IVR system comprises one of speech, DTMF, and both, and wherein the output modality comprises speech.

45. (Original) The system of claim 39, wherein CML is implemented using a declarative format.

46. (Canceled)

47. (Original) The system of claim 45, wherein the declarative format comprises an extensible markup language (XML) format.

48. (Original) The system of claim 39, wherein CML is implemented using a combination of declarative and imperative components.

49. (Original) The system of claim 39, wherein the I/O events are multi-modal.

50. (Original) The system of claim 39, wherein the I/O events are speech only.

51. (Original) The system of claim 39, wherein the I/O events comprise one of synthesized speech output, playback of audio files, recognition of spoken input, recognition of DTMF (dual tone multifrequency) input, capturing of spoken input, capturing of text input, and a combination thereof.

52. (Original) The system of claim 39, wherein the conversational browser one of comprises a CVM (conversational virtual machine) and executes on top of a CVM.

53. (Original) The system of claim 39, wherein the conversational browser and platform comprise an embedded architecture that operates in a pervasive computing client device.

54. (Original) The system of claim 39, wherein the content server, conversational browser and platform one of execute on a single machine and are distributed over a communications network.

55. (Original) The system of claim 39, wherein standard networking protocols are utilized for accessing CML content pages and applications from the content server.

56. (Original) The system of claim 55, wherein the standard networking protocols comprise one of TCP/IP (transmission control protocol/Internet protocol), HTTP (hypertext transmission protocol), WAP (wireless application protocol), VoIP (voice over internet protocol), distributed network protocols, *ad hoc* network protocols, hopping network protocols, and a combination thereof.

57. (Original) The system of claim 39, further comprising a transcoder for converting legacy information formats of the content server into CML based on predefined transcoding rules.

58. (Original) The system of claim 57, wherein the transcoder interprets backend logic of the content server to generate a corresponding conversational dialog structure.

59. (Original) The system of claim 57, wherein the transcoder executes on one of the conversational browser, a conversational proxy server, the content server, and distributed among combination thereof.

60. (Original) The system of claim 57, wherein the conversational browser comprises a registration mechanism for registering one of its conversational capabilities, conversational state, and both, with one of the content server, transcoder and both.

61. (Original) The system of claim 40, wherein the registration mechanism comprises one of a cookie, a form, procedural protocols, and a combination thereof.

62. (Original) The system of claim 60, wherein the transcoder generates one of a custom CML file and CML application based on the conversational capabilities of the conversational browser.

63. (Original) The system of claim 39, wherein the CML comprises a plurality of capability-based frames, each providing different dialog levels based on the conversational capabilities of the conversational browser.

64. (Original) The system of claim 39, wherein CML comprises an active link that is one of spoken to and hidden from the user.

65. (Original) The system of claim 39, wherein CML comprises a link to conversational data files for processing user input.

66. (Original) The system of claim 39, wherein CML comprises a link to at least one distributed conversational engine for processing user input.

67. (Original) The system of claim 39, wherein CML comprises a link to an audio file for playback to a user.

68. (Original) The system of claim 39, wherein CML comprises a confirmation request tag for confirming user input.

69. (Original) The system of claim 39, wherein CML comprises TTS (text-to-speech) markup for specifying audio playback of text.

70. (Original) The system of claim 39, wherein CML comprises one of scripting language and imperative code for executing conversational applications that direct their own dialog.

71. (Original) The system of claim 39, wherein CML comprises a link to one of a plug-in, an applet, a dialog object, and a combination thereof, for executing a conversational task.

72. (Original) The system of claim 39, wherein CML comprises one of (1) a top level element that groups other CML elements; (2) an element that specifies output to be spoken to the user (3) a menu element for encapsulating a menu that presents the user with a list of choices, wherein each choice is associated with a target address identifying a CML element to visit if the corresponding choice is selected; (4) a form element for encapsulating a form that allows the user to input at least on item of information and transmit the at least one item of information to a target address; and (5) a combination thereof.

73. (Original) The system of claim 72, wherein a target address comprises one of a URL (uniform resource locator), a relative URL, a socket address, and a protocol identifier that identifies a desired protocol for establishing communication.

74. (Original) The system of claim 73, wherein the target address is used for accessing one of a local and distributed conversational service.

75. (Original) The system of claim 73, wherein the target address is used for accessing one of a local and distributed conversational data file.

76. (Original) The system of claim 72, wherein the form comprises input fields, and wherein a grammar is associated with each input field to specify appropriate responses for filling the corresponding input field.

8

77. (Original) The system of claim 76, wherein each choice of the menu is associated with a grammar.

78. (Original) The system of claim 77, wherein the menu and form comprise an attribute that specifies when their corresponding grammars are active.

79. (Original) The system of claim 72, wherein the conversational browser employs a mixed initiative dialog feature.

80. (Previously Presented) A method for accessing information, comprising the steps of:

(a) processing an input command with at least one of a plurality of conversational engines;

(b) generating a request based on the processed input command to access a CML (conversational markup language) file from a content server, the CML file comprising meta-information to implement a conversational dialog in a plurality of user interface modalities;

(c) transmitting the request and accessing the requested CML file from a content server using a standard networking protocol; and

(d) processing the meta-information comprising the CML file to render the conversational dialog in one or more of the plurality of user interface modalities.

81. (Original) The method of claim 80, wherein steps (b), (c) and (d) are executed by a conversational browser.

82. (Original) The method of claim 81, wherein the conversational browser executes on top of an operating platform of a pervasive computing device.

83. (Original) The method of claim 80, wherein steps (a) and (d) are one of executed locally using a conversational engine and conversational arguments, executed remotely using distributed conversational engines and conversational arguments, and a combination thereof.

84. (Original) The method of claim 81, further comprising the step of:

registering one of conversational capabilities, conversational state, and both of the conversational browser with the content server.

85. (Original) The method of claim 84, further comprising the step of customizing the CML file based on the conversational capabilities of the conversational browser..

86. (Original) The method of claim 80, further comprising the step of transcoding legacy content of the content server into CML based on predefined transcoding rules.

87. (Original) The method of claim 86, wherein the step of transcoding is performed by one of a conversational browser, a transcoding proxy associated with a content server from which the legacy content is accessed, and a combination thereof.

88. (Original) The method of claim 86, further comprising the step of processing backend logic of the content server to generate a corresponding conversational dialog structure.

89. (Original) The method of claim 80, wherein step (d) comprises the step of playing back an audio file, generating synthesized speech output, and a combination thereof.

90. (Original) The method of claim 80, wherein CML is implemented using a declarative format.

91. (Original) The method of claim 90, wherein the declarative format is based on XML (extensible markup language).

92. (Canceled)

93. (Original) The method of claim 80, wherein CML comprises declarative and procedural components.

94. (Original) The method of claim 80, wherein CML comprises one of (1) a top level element that groups other CML elements; (2) an element that specifies output to be spoken to the user (3) a menu element for encapsulating a menu that presents the user with a list of choices, wherein each choice is associated with a target address identifying a CML element to visit if the corresponding choice is selected; (4) a form element for encapsulating a form that allows the user to input at least on item of information and transmit the at least one item of information to a target address; and (5) a combination thereof.

95. (Original) The method of claim 94, wherein a target address comprises one of a URL (uniform resource locator), a relative URL, a socket address, and a protocol identifier that identifies a desired protocol for establishing communication.

96. (Original) The method of claim 95, wherein step (d) comprises the step of accessing one of a local and distributed conversational service associated with the target address.

97. (Original) The system of claim 94, wherein step (d) comprises the steps of:
audibly outputting an introductory message of a form to a user, if a form element is visited;
activating a grammar for each input field of the form, wherein the grammar specifies at least one appropriate value for the corresponding input field; and
filling a field if the user speaks an appropriate response for the field.

98. (Original) The method of claim 97, further comprising the steps of:
audibly outputting an introductory message of a menu to a user, if a menu element is visited;

11

audibly outputting prompt text of choice elements associated with the menu;

activating a grammar for each choice element, wherein the grammar specifies a response for selecting the given choice element; and

selecting a choice element if the spoken response of the user matches the grammar associated with the choice element.

99. (Original) The method of claim 98, further comprising the step of activating the grammars of the menu and form for a predefined duration based on an attribute value specified by the menu and form elements.

100. (Original) The method of claim 94, further comprising the step of employing a mixed initiative dialog feature for one of filling form input fields and selecting menu choices.